Event Registration Mobile Application

1 Required Software Engineering Tools

- Android Studio
  (http://developer.android.com/sdk/index.html)

Visit the above URL to download and install the Android Studio IDE.

On Intel processors, you may have to install HAMX individually (https://software.intel.com/en-us/android/articles/installation-instructions-for-intel-hardware-accelerated-execution-manager-windows), which may require enabling virtualization in your BIOS. You may also have to experiment with the best setting for RAM allocation needed by HAMX (the more, the better). Note that other virtualization environments may need to be disabled on your machine (e.g. Docker).

2 Description of Event Registration Application

After delivery of our first release, we have discovered that there was an additional requirement:

- The Event Registration application shall provide the ability to add a participant by specifying the participant's name.
- The Event Registration application shall provide the ability to add an event by specifying the event's name, date, start time, and end time.
- The Event Registration application shall provide the ability to register a participant to an event.
- The Event Registration application shall run on the Android mobile platform.

3 Exporting a JAR Archive from Eclipse

To export your Java Swing project from Eclipse into a reusable JAR archive, see
https://ecse321-winter2017-mcgill.github.io/EventRegistration-Tutorials/#_exporting_the_java_project

4 Tasks of the Assignment

To complete this assignment, you need to follow the tutorial at
https://ecse321-winter2017-mcgill.github.io/EventRegistration-Tutorials/#_android

You need to complete all steps of **Section 2**. Note that the tutorial is less detailed this time, i.e. you need to figure out some of the steps on your own!

Submission

This second part of the assignment is to be done by the same teams as the first part. Your team is required to hand in a ***single zip file*** of the Android project with your implementation by ***Wednesday, February 1, 2017 at 23:30***. To create the zip file, simply zip up the folder of your Android project with your favorite archiving tool. If you realize that you need to make changes to your submission, do not resubmit only the file(s) that have changed, but rather resubmit another complete zip file.

Note that this is only part 2 of 3 parts. The third and final part of assignment #1 is posted together with this assignment on January 23. Each part is about the same amount of work and **all parts are due on Wednesday, February 1, 2017**. Do not wait until the very end to complete this assignment. Start early.

Each team member must make contributions to the assignment. A team member who does not contribute to the assignment receives a mark of 0 for the assignment. A team member may optionally email a confidential statement of work to the instructor before the due date of the assignment. A statement of work first lists in point form the parts of the assignment to which the team member contributed. In addition, the statement of work also describes whether the work load was distributed fairly evenly among the team members. A statement of work may be used to adjust the mark of a team member who is not contributing sufficiently to the assignment. It is not necessary to send a statement of work, if a team distributed the work for the assignment fairly evenly and each team member contributed sufficiently.

Marking Scheme

Part of Assignment	***Marks***
Use of Jar File from Desktop Application	30
Adherence to Model-View-Controller Pattern (only View is replaced)	20
Implementation of Functionality:	40
a) Add Participant	10/40
b) Add Event	10/40
c) Register	10/40
d) Persistence	10/40
Implementation of Validation Checks for:	10
a) Add Participant	2/10
b) Add Event	4/10
c) Register	4/10
Total Marks:	100
The total mark may be adjusted based on the actual contributions of a team member to the assignment.	